

U.K. PET USERS CLUB

NEWSLETTER NO. 0

Welcome to the U.K. PET Users Club and in this issue, we have some hints on graphics animation, details of BASIC interpreter timings and memory usage, and of ASCII codes plus much more.

Printed in this and all subsequent issues, there will be a list of software available through the Users Club. The original list is small and Commodore Systems generated, but we hope to add many programmes in the near future. We intend to form a "Common Library" which will contain programmes submitted by PET users and we hope you will contribute. The contents of both Libraries will be printed in each issue.

For each of your programmes accepted, you will be given the choice of up to three from the Common Library, free of charge or you may purchase any programme from the Common Library for £1.00 per programme plus 30 pence for p+p. We also hope that you will write to us telling us of your applications for and your experiences with the PET. We will endeavour to publish as many of your letters and comments as we can and if anyone feels like submitting a small article, this will be most welcome. If you require a personal reply this will be guaranteed by enclosing a self addressed, stamped envelope.

The address to which you send your correspondence or orders is:-

The Software Manager,  
Commodore Systems Division,  
360, Euston Road,  
London, N.W.1.

N.B. Please note CHANGE OF OUR ADDRESS.



### A SHORT NOTE ON MOVING THINGS

Difficulty has been experienced by some of our users in moving the cursor under programme control and questions have been asked about how graphs and plots such as sine curves can be displayed.

As you probably already know, cursor control characters in quotes when printed will cause the cursor to move accordingly. If your experiments so far with this technique are giving slightly odd results, do make sure that you are terminating your print statement with a semi-colon. Failure to do this will cause the machine to output a carriage return/line feed at the end of the print statement, leaving the cursor just where you don't want it.

On the subject of plotting, the basic trick used with this type of graphic system is to first create strings full of cursor movement characters and then access parts of them using MID\$, etc. dynamically.

For a really good example of the use of this technique, I suggest you have a look at the technique used in our biorhythm programme.

## BITS AND PIECES

### ARE YOU READY?

There have been reported mysterious occurrences of the out of data error when editing and fiddling about in general.

This is not a bug, but is due to pressing RETURN whilst the cursor is over the READY prompt. The machine interprets this as READ Y and as there is usually no corresponding data statement around we get the error.

### REDO

It must be remembered that when RETURN is pressed, the machine consumes everything on the same line as the cursor, so even if you have correct information at the beginning of a given line, a single character of an incorrect type far over on the right hand side of the screen on the same line is likely to cause problems. A rather problematical example of this situation occurs if you try and put up a graphic form or set of boxes on the screen and then under programme control ask for data with an input statement, e.g.

### NUMBER ?

When the number is typed and RETURN is pressed, the graphics character making up the right-hand side of the box will be entered as part of the inputting data. In the case of input to a numeric variable, the graphics character is of course non-numeric and not allowed and will give the error ? Redo from start, so you must always leave such boxes open ended.

## INVERSE TRIGNOMETRIC FUNCTIONS

Here are a couple of handy methods of obtaining arc sine and arc cosine (remember, the result will be in radians).

ASN X = ATN (X/SQR(1-X<sup>2</sup>))

ACS X = ATN (SQR(1-X<sup>2</sup>)/X)

For those of you who are used to working in degrees, here are some handy user defined functions:

DEFFNS(V) = SIN(V/(180/π))

DEFFNC(V) = COS(V/(180/π))

DEFFNT(V) = TAN(V/(180/π))

These are three user defined functions which when called with arguments and degrees will give the appropriate results. In these examples V can be any variable but if all three are defined in the same programme, you must use three different dummy variables.

EXAMPLE:           PRINT FNS(30)

Result of this will be .5. Notice that the argument for FNS, or FN anything for that matter, can be either a variable or numeric constant. Also, after a programme containing these definitions has been run, these functions may be called using FN in the direct mode, that is, from the keyboard directly without being in a programme.

## SECOND CASSETTE UNIT

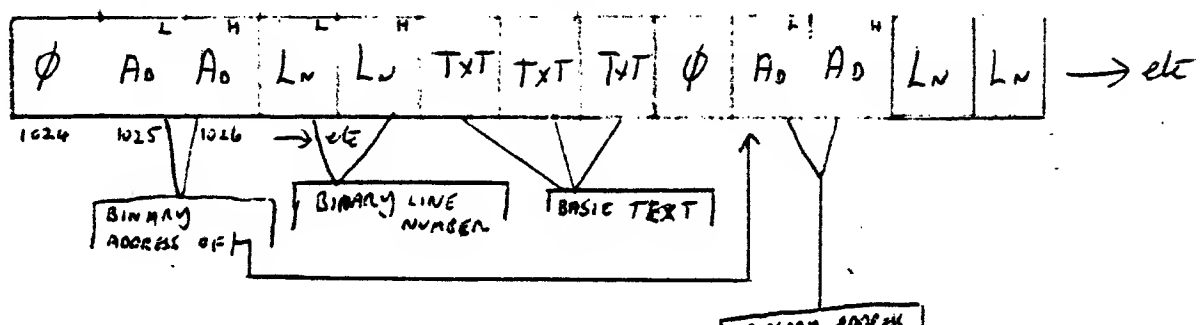
The second cassette unit with lead and plug ready to go straight onto the second cassette port on the rear of the PET, will be available from the beginning of May at a cost £55.00 plus V.A.T. Some of the early units shipped, may have a small section of the body cut away at the back. This is because, these particular cassettes were meant originally to fit inside the PET.

## 6502 PROGRAMMING MANUALS

This really excellent MOS technology generated 220 page manual will be available at a cost of £5.00 plus 30pence p+p from the beginning of May. This manual, if read carefully from the beginning, is an excellent guide to microprocessor programming in general, as well as being a first rate 6500 reference book. It is worth mentioning here, that for those of you who are interested in developing their knowledge and expertise in the field of microprocessing and cannot, at this point of time, afford a PET, then they might, perhaps, look at a KIM, a valuation board, which sells for £149 and contains a keyboard, display, 2K ROM plus 1K RAM, cassette interface, 2 timers and much more. This valuation board is available in London through A.E. Marshalls Ltd or in the West Country from G.R. Electronics of Newport, Gwent.

## BASIC PROGRAMME STORAGE

Below is a diagrammatic representation of the method used by the CBM basic interpreter for storing programmes in memory.



## EDITING

There is an interesting property of the screen edit routine which gives rise to the following effects:-

If you insert using the INS key, more spaces than you type in characters, the DEL key must be pressed twice the number of times there are spare spaces. E.g. If you insert six spaces in a middle of a line and only type in four new characters, the first two presses of the DEL key will produce inverse characters which will disappear on the next two presses. Remember, the INS key will move all characters including the one under the cursor to the right, whilst the DEL key will delete the character on its immediate left.

## INTERRUPT STRUCTURE

Interrupts (including Break or Software Interrupts) are handled by software polling.

When the processor recognizes an interrupt it vectors through FFFE, FFFF in ROM to a routine that first inspects the processor hardware (IRQ line low).

If it was caused by a Break instruction, a Jump Indirect is executed through locations 021B, C. If by a hardware interrupt then a Jump Indirect is taken through locations 0219, A.

These locations being in RAM may be user-modified to point to extra user code ahead of normal interrupt processing.

Note, however that the IRQ pointer is used by the cassette routines and should be restored to standard values before the cassette Save or Load functions are called.

Various sections of the I/O chips can be set up to cause interrupts through the IRQ line.

Example: POKE 59470,2 enables a negative edge on the user port CAL line to cause an interrupt.

However, have your code set up to handle it when it happens!

Also note that each pass through the regular interrupt code increments the time register.



# PET Matrix-Decoded Keyboard

See 515 & 516 in table below

|    | 8  | 7  | 6  | 5  | 4  | (3) | 2 | 1 |     |    |    |    |   |    |   |
|----|----|----|----|----|----|-----|---|---|-----|----|----|----|---|----|---|
| 64 | I  | "  | #  | \$ | %  | '   | & | \ | ( ) | +  | ho | ↑↑ | ↵ | de |   |
| 48 | Q  | W  | E  | R  | T  | Y   | U | I | O   | P  | ↑  | 7  | 8 | 9  | / |
| 32 | A  | S  | D  | F  | G  | H   | J | K | L   | :  |    | 4  | 5 | 6  | * |
| 16 | Z  | X  | C  | V  | B  | N   | M | , | :   | ?  | re | 1  | 2 | 3  | + |
| 0  | sh | rv | @  | [  | ]  | sp  |   | < | >   | sh | sh | 0  | . | -  | = |
|    | 16 | 15 | 14 | 13 | 12 | 11  |   |   |     |    |    | 10 | 9 |    |   |

## Interesting Locations Accessible from BASIC

### Location (decimal)

### Contents

|   |   |
|---|---|
| <sup>H</sup><br><sup>L</sup><br>225, 224<br>226   | Byte address of screen line with Cursor<br>Character position of Cursor (0 to 79)   |
| 515   | Matrix-coordinate (row+column) of last key down<br>255 if no key down   |
| 516   | 1 if shift down, 0 if shift up  |
| 525<br>526-534                                    | No. of characters in Keyboard Buffer<br>Keyboard Buffer   |
| 578 to 587<br>588 to 597<br>598 to 607<br><br>610 | Logical numbers of open files<br>Device numbers of open files<br>Read/write modes of open files<br><br>How many open files  |
| 512, 513, 514<br>518, 517<br>59465, 59464         | Clock that increments 60 times a second<br>Clock that increments 30 times a second?<br>Clock that decrements every microsecond  |
| 59456   | WAIT 59456,32,32 waits for vertical retrace of display  |
| 64824   | SYS(64824) simulates power-on reset   |
| 59469   | Interrupt Flag Register; e.g., to input user port CA1:<br>I=PEEK(59469) AND 2: POKE 59469,I: IF I=0 THEN CA1 low  |
| 59411   | IEEE PIA B Control, e.g., to run cassette#1 motor N jiffies:<br>100 POKE 59411,53: T=TI<br>200 IF TI-T<N GOTO 200<br>300 POKE 59411,61<br>ADVICE: Run motor at least 3 jiffies per 191 output chars |

# TIMING TABLES

## BASIC STATEMENTS AND I/O

| CONSTRUCT               | APPROX. TIME (MILLISEC)    | STRING FUNCTIONS (Cont'd) | APPROX. TIME (MILLISEC) |
|-------------------------|----------------------------|---------------------------|-------------------------|
| FRE                     | 1 to 10                    | VAL                       | 1.3                     |
| PEEK, POKE              | 1                          | =, <, >, <=, >, >=        | 3 to 4                  |
| TI\$                    | 3 to 4                     |                           |                         |
| TI                      | 1                          | ARITHMETIC FUNCTIONS      |                         |
| GET                     | 1 to infinity              | FUNCTION ..               | APPROX. TIME (MILLISEC) |
| POS                     | 1                          | ABS                       | 0.6                     |
| PRINT X or PRINT        | 15 to 19                   | ATN                       | 42                      |
| PRINT X\$;              | 14 + LEN (X\$)/2           | COS                       | 27                      |
| READ X and DATA 3       | 9                          | EXP                       | 27                      |
| REM                     | 0.2 to 2                   | INT                       | 1.2                     |
| RESTORE                 | 0.3                        | LOG                       | 23                      |
| TAB                     | 2                          | RND RND (-1)              | 1.0                     |
| SPC(N)                  | 1 + 0.6*N                  | RND (0)                   | 0.9                     |
| FOR I = ... NEXT 1      | 4.0 + (1.6 each)           | RND (1)                   | 4.1                     |
| STEP                    | 1.3                        | SGN                       | 1.1                     |
| IF                      | 0.4                        | SIN                       | 25                      |
| GOTO or GOSUB           | 1.1                        | TAN                       | 50                      |
| ON A GOTO or GOSUB      |                            | user FN                   | 2.4                     |
| L1 ..... L <sub>M</sub> | 0.5 + (0.3*A)<br>+ (0.2*M) |                           |                         |
| RETURN                  | 0.9                        | ARITHMETIC OPERATORS      |                         |

Using colon,:, saves 0.6 over new line.

SAVE or LOAD

15 sec + (2 sec per 100 char)

i.e. 500 baud.

## STRING FUNCTIONS

| FUNCTION        | APPROX. TIME (MILLISEC) |
|-----------------|-------------------------|
| +               | 0.5 + (0.2 per char)    |
| ASC             | 1                       |
| CHR\$           | 1.2                     |
| LEFT\$, RIGHT\$ | 3 + (0.025 per char)    |
| LEN             | 0 to 8                  |
| MID\$           | 4 + (0.025 per char)    |
| STR\$           | 7 to 10                 |

| SYMBOL             | APPROX. TIME (MILLISEC) |
|--------------------|-------------------------|
| OfB, 1fB           | 0.3                     |
| 2fB                | 32                      |
| else               | 50 to 100               |
| / O/B, A/1         | 0.5                     |
| else               | 2 to 5                  |
| * O*B, A*O         | 0.4                     |
| else               | 1.5 to 3                |
| +                  | 0.3 to 1                |
| -                  | 0.3 to 1                |
| =, <, >, <=, >, >= | 0.7                     |
| AND, OR            | 1.7                     |
| NOT                | 1.4                     |

# VARIABLES AND CONSTANTS

## ITEM

A, A\$, A =, A\$ =

AA, AA\$, AA =, AA\$ =

A%

A% =

999

.999

E16

E-16

"ABCDE"

M (I, J, ....)

## APPROX. TIME (MILLISEC)

0.7 to  $(0.7 + nv \cdot 0.1)$

nv = no. of variables

in program

0.2 more than above

0.3 more than A

0.6 more than A =

1 per digit

0.7 + (4.2 per digit)

0.2 + (0.4\* exponent)

0.2 + (3.0\* exponent)

(0.6 to 0.7) + (0.02

per char)

(1 to 1.5)\*

(no. of subscripts)

# TIMING PROGRAM

100 N = 300

200 T1 = T1

300 FOR I = 1 TO N

400 REM PUT TEST CONSTRUCT HERE

500 NEXT I

600 T2 = T1

700 FOR I = 1 TO N

800 NEXT I

900 T3 = T1

1000 PRINT 1000\* (2\*T2-T1-T3) / (60\*N)

1100 END

## MEMORY USAGE (IN BYTES)

BASIC 1028 (I/O buffers, tables etc)

each statement

4 for line number and following space,

regardless for the line number

1 for each BASIC keyword

1 for each other character, including

RETURN

each variable with a value assigned, regard

less of spelling or value takes 7 bytes;

for string variables, add the length of

the string

each array (N.B., size includes 0th element)

take  $f \cdot (\text{size} + 1) + (2 \text{ per dimension})$  where

$f=5$  for floating point arrays,  $f=2$  for

integer arrays, and  $f=3$  for string arrays.

The system slows down noticeably when memory is

neary full.

| Character | ASC/CHR | PEEK/POKE | Character | ASC/CHR | PEEK/POKE | Character | ASC/CHR | PEEK/POKE | Character | ASC/CHR | PEEK/POKE |
|-----------|---------|-----------|-----------|---------|-----------|-----------|---------|-----------|-----------|---------|-----------|
|           | 0       |           | @         | 64      | 0         |           | 128     |           |           | 192     | 64        |
|           | 1       |           | A         | 65      | 1         |           | 129     |           | ,a        | 193     | 65        |
|           | 2       |           | B         | 66      | 2         |           | 130     |           | ,b        | 194     | 66        |
|           | 3       |           | C         | 67      | 3         |           | 131     |           | ,c        | 195     | 67        |
|           | 4       |           | D         | 68      | 4         |           | 132     |           | ,d        | 196     | 68        |
|           | 5       |           | E         | 69      | 5         |           | 133     |           | ,e        | 197     | 69        |
|           | 6       |           | F         | 70      | 6         |           | 134     |           | ,f        | 198     | 70        |
|           | 7       |           | G         | 71      | 7         |           | 135     |           | ,g        | 199     | 71        |
|           | 8       |           | H         | 72      | 8         |           | 136     |           | ,h        | 200     | 72        |
|           | 9       |           | I         | 73      | 9         |           | 137     |           | ,i        | 201     | 73        |
|           | 10      |           | J         | 74      | 10        |           | 138     |           | ,j        | 202     | 74        |
|           | 11      |           | K         | 75      | 11        |           | 139     |           | ,k        | 203     | 75        |
|           | 12      |           | L         | 76      | 12        |           | 140     |           | ,l        | 204     | 76        |
| RETURN    | 13      |           | M         | 77      | 13        | RETURN    | 141     |           | ,m        | 205     | 77        |
|           | 14      |           | N         | 78      | 14        |           | 142     |           | ,n        | 206     | 78        |
|           | 15      |           | O         | 79      | 15        |           | 143     |           | ,o        | 207     | 79        |
|           | 16      |           | P         | 80      | 16        |           | 144     |           | ,p        | 208     | 80        |
| ↓         | 17      |           | Q         | 81      | 17        | ↑         | 145     |           | ,q        | 209     | 81        |
| RVS       | 18      |           | R         | 82      | 18        | RVSoff    | 146     |           | ,r        | 210     | 82        |
| HOME      | 19      |           | S         | 83      | 19        | CLEAR     | 147     |           | ,s        | 211     | 83        |
| DEL       | 20      |           | T         | 84      | 20        | INST      | 148     |           | ,t        | 212     | 84        |
|           | 21      |           | U         | 85      | 21        |           | 149     |           | ,u        | 213     | 85        |
|           | 22      |           | V         | 86      | 22        |           | 150     |           | ,v        | 214     | 86        |
|           | 23      |           | W         | 87      | 23        |           | 151     |           | ,w        | 215     | 87        |
|           | 24      |           | X         | 88      | 24        |           | 152     |           | ,x        | 216     | 88        |
|           | 25      |           | Y         | 89      | 25        |           | 153     |           | ,y        | 217     | 89        |
|           | 26      |           | Z         | 90      | 26        |           | 154     |           | ,z        | 218     | 90        |
|           | 27      |           | [         | 91      | 27        |           | 155     |           |           | 219     | 91        |
|           | 28      |           | \         | 92      | 28        |           | 156     |           |           | 220     | 92        |
|           | 29      |           | ]         | 93      | 29        | ←         | 157     |           |           | 221     | 93        |
| ⇒         | 30      |           | ↑         | 94      | 30        |           | 158     |           |           | 222     | 94        |
|           | 31      |           | →         | 95      | 31        |           | 159     |           |           | 223     | 95        |
| space     | 32      | 32        | → space   | 96      | 32        |           | 160     |           |           | 224     | 96        |
| !         | 33      | 33        | !         | 97      | 33        |           | 161     | 97        |           | 225     | 97        |
| "         | 34      | 34        | "         | 98      | 34        |           | 162     | 98        |           | 226     | 98        |
| #         | 35      | 35        | #         | 99      | 35        |           | 163     | 99        |           | 227     | 99        |
| \$        | 36      | 36        | \$        | 100     | 36        |           | 164     | 100       |           | 228     | 100       |
| %         | 37      | 37        | %         | 101     | 37        |           | 165     | 101       |           | 229     | 101       |
| &         | 38      | 38        | &         | 102     | 38        |           | 166     | 102       |           | 230     | 102       |
| '         | 39      | 39        | '         | 103     | 39        |           | 167     | 103       |           | 231     | 103       |
| (         | 40      | 40        | (         | 104     | 40        |           | 168     | 104       |           | 232     | 104       |
| )         | 41      | 41        | )         | 105     | 41        |           | 169     | 105       |           | 233     | 105       |
| *         | 42      | 42        | *         | 106     | 42        |           | 170     | 106       |           | 234     | 106       |
| +         | 43      | 43        | +         | 107     | 43        |           | 171     | 107       |           | 235     | 107       |
| ,         | 44      | 44        | ,         | 108     | 44        |           | 172     | 108       |           | 236     | 108       |
| -         | 45      | 45        | -         | 109     | 45        |           | 173     | 109       |           | 237     | 109       |
| .         | 46      | 46        | .         | 110     | 46        |           | 174     | 110       |           | 238     | 110       |
| /         | 47      | 47        | /         | 111     | 47        |           | 175     | 111       |           | 239     | 111       |
| 0         | 48      | 48        | 0         | 112     | 48        |           | 176     | 112       |           | 240     | 112       |
| 1         | 49      | 49        | 1         | 113     | 49        |           | 177     | 113       |           | 241     | 113       |
| 2         | 50      | 50        | 2         | 114     | 50        |           | 178     | 114       |           | 242     | 114       |
| 3         | 51      | 51        | 3         | 115     | 51        |           | 179     | 115       |           | 243     | 115       |
| 4         | 52      | 52        | 4         | 116     | 52        |           | 180     | 116       |           | 244     | 116       |
| 5         | 53      | 53        | 5         | 117     | 53        |           | 181     | 117       |           | 245     | 117       |
| 6         | 54      | 54        | 6         | 118     | 54        |           | 182     | 118       |           | 246     | 118       |
| 7         | 55      | 55        | 7         | 119     | 55        |           | 183     | 119       |           | 247     | 119       |
| 8         | 56      | 56        | 8         | 120     | 56        |           | 184     | 120       |           | 248     | 120       |
| 9         | 57      | 57        | 9         | 121     | 57        |           | 185     | 121       |           | 249     | 121       |
| :         | 58      | 58        | :         | 122     | 58        |           | 186     | 122       |           | 250     | 122       |
| ;         | 59      | 59        | ;         | 123     | 59        |           | 187     | 123       |           | 251     | 123       |
| <         | 60      | 60        | <         | 124     | 60        |           | 188     | 124       |           | 252     | 124       |
| =         | 61      | 61        | =         | 125     | 61        |           | 189     | 125       |           | 253     | 125       |
| >         | 62      | 62        | >         | 126     | 62        |           | 190     | 126       |           | 254     | 126       |

| <u>Price</u> | <u>Name</u>          | <u>Description</u>   |
|--------------|----------------------|--|
| £ 8.00       | Othello              | Cunning game of skill. Two levels of play, you against the computer.   |
| £ 6.00       | Pontoon              | Board game. True 52 card pack plus amazing graphics.   |
| £ 8.00       | Wrap Trap            | Dynamic graphics game in which the player has to trap the computer. Good arcade quality graphics.  |
| £ 3.00       | Noughts and Crosses  | You against the computer.  |
| £ 8.00       | Lunar Lander         | First class game of skill - in real time and with the odd surprise!  |
| £ 5.00       | Rotate               | Difficult if you are not an expert! Similar to little plastic trays with moveable letters and letter missing.                              |
| £ 8.00       | Biorhythms           | Carefully written programme. Good graphics, with a real plot routine. Great fun for all the family.  |
| £15.00       | Disassembler         | 6500 series full disassembler asks for decimal starting location and lists from this point, gives full mnemonics and handles ASCII tables. |
| £ 3.00       | Machine Code Handler | This programme allows you to type in a list of HEX codes from a given location. These routines can then be called using the SYS verb.      |

Please send 50p for p + p.

There are many more programmes in preparation including mathematics statistics and even a line re-numbering routine. These and many more will be announced in our next Newsletter. Many people have asked about a series of good quality short cassettes for the PET, so we have decided to make available C12 cassettes, with blank labels and in individual boxes at a cost of 50 pence each. There is a minimum quantity order of 10 cassettes, due to the small value of this item. Terms must be C.W.O.

It is also worth mentioning that Compendium Books of Camden Town have an excellent Home Computing section - carrying such things as Dr. Dobbs and Byte etc. Nearest tube is Camden Town also 74B and 31 busess.

#### STOP PRESS

R.Bailey Associates of 31 Bassett Road, London, W.10 are offering the following 2 items for sale from the beginning of May.

#### Item No. 1.

I EEE to R S 232 Slash Serial Interface for £79.50 + V.A.T.

#### Item No. 2.

Plug in Memory complete with Power Supply:

8K:- £225 + V.A.T.

16K:- £400 + V.A.T.